

A COMPOSITE STRUCTURED/UNSTRUCTURED-MESH EULER METHOD FOR COMPLEX AIRFOIL SHAPES

by

H. Hefazi*
Aerospace Engineering Department
California State University, Long Beach

and

L. T. Chen**
Douglas Aircraft Company
Long Beach, California

N 93-27489
160472
P. 6

Abstract

A general two-dimensional Euler zonal method has been developed for computing flows about complex airfoil geometries such as multielement and iced airfoils. The method utilizes a composite structured and unstructured grid generated using conformal mapping and Delaunay triangulation, respectively. The finite-volume Euler method is then modified to couple solutions in the zones with structured and unstructured grids. Solutions about an iced airfoil and a multielement airfoil are given as examples of applications of the scheme.

1.0 Introduction

The aerodynamic analysis of complex airfoil sections continues to receive much attention in experimental, theoretical and computational studies. Such analyses of airfoil flows include iced airfoils, multielement airfoils, and advanced airfoil concepts with divergent trailing edges. Conventional computational methods based on a single zone mesh, developed for simple airfoil configurations, are generally not suitable for these geometries since a single structured grid of sufficient quality cannot be generated. Zonal or unstructured-mesh methods that can provide adequate mesh resolution near high pressure gradient regions in each zone are needed to handle the complexity of the flowfields. An attractive zonal approach is to use structured meshes in most parts of the domain and unstructural meshes in enclosed regions next to the portions of airfoil that are difficult to model with structured grids. The objective of this paper is to develop a general two-dimensional Euler method based on this approach.

Although many methods have been developed for the generation of structured grids around simple geometries, few can be readily extended to complicated two- and three-dimensional shapes such as multielement or iced airfoils or multicomponent aircraft configurations. Also, in many applications of structured grids, the quality of the generated mesh is not uniform throughout the computational domain, resulting in poor resolution in specific regions of importance. For this reason, triangular (tetrahedral) meshes have proven very attractive, since relatively complex geometries can be meshed efficiently, and an almost arbitrary degree of mesh adaption and refinement can be achieved by the addition of control points. Furthermore, the development of flow algorithms that

do not depend on the inherent structure of the grid points have eliminated the restriction of grid structure and made triangles and tetrahedrons suitable shapes with which to resolve complicated geometric regions.

The debate over relative advantages of structured and unstructured grid methods is an ongoing matter. While unstructured grid methods have the clear advantage in that they can treat complicated problems such as a complete aircraft configuration under static or dynamic deformation,¹ they have been regarded as less efficient and less accurate than their structured counterparts.^{2,3} Lack of robust acceleration techniques, such as multigrid schemes for three-dimensional problems, is also a clear disadvantage. Furthermore, direct and implicit solvers for unstructured grids are developed, but in general are not as efficient and effective as their structured counterparts.^{4,5} Structured flow solvers have many other highly desirable features including efficient grid generation techniques and smaller computer time and memory requirements.

An efficient way of analyzing a complex geometry of several components is through a zonal approach, using composite structured and unstructured grids. This approach requires considerably less memory than using an entire unstructured mesh capable of handling the same geometry. In view of the different desirable features of structured and unstructured meshes, the present zonal approach takes advantage of structured meshes in appropriate regions while using the versatility of the unstructured grids in others.

In this paper, a general two-dimensional zonal boundary interactive scheme utilizing combinations of structured and unstructured mesh types is presented. Solutions for flow about multielement and iced airfoils are given as an example of the application of the scheme.

2.0 Grid Generation

A brief grid generation process will be described in this section. A base structured grid which encompasses the entire flowfield is first generated using existing methods. Regions of undesirable grid quality are then identified and categorized as subsequent unstructured-mesh zones. Triangular grids can be generated in these regions using existing mesh points and additional points as necessary. An unstructured grid generation method based on Delaunay triangulation, developed earlier,⁶ is applied for this purpose. An essential requirement for obtaining satisfactory meshes when using the Delaunay triangulator is

*Associate Professor.

**Senior Principal Engineer/Scientist.

appropriate placement of interior points, since this scheme gives no guidance on where to place the mesh points. Inappropriate placement of mesh points results in poor quality meshes, even though they satisfy the Delaunay criteria. Therefore, certain criteria for introducing interior mesh points have to be established before applying the Delaunay triangulation.

The Delaunay triangulation method is applied in three steps to generate triangular meshes in the unstructured zone enclosed by the surrounding structured meshes and solid surface boundary. The first step is to triangulate the points along the body and zonal boundaries in order to obtain an initial triangulation based solely on existing boundary points.

The second step deals with the placement of interior points. It is important to place a sufficiently dense mesh of points in high gradient regions such as corner regions, leading- and trailing-edge regions, etc. A combination of C-mesh points around the leading edge and Cartesian mesh points in other parts of the domain can be chosen as the interior points. Based on Bowyer's algorithm,⁷ a series of new points are added one by one to the existing triangulation, removing triangles close to the point being inserted, and reconnecting the new point to the existing nodes in such a way as to form new triangles which satisfy the Delaunay criteria. This procedure is repeated for all new points introduced in the domain.

The final phase is a grid-smoothing procedure. This is necessary because points that are introduced sometimes fall too close to each other, resulting in skewed triangles, such as triangles with large aspect ratios. An iterative Laplacian smoothing scheme is applied to improve the overall quality of the unstructured mesh.

3.0 Finite Volume Scheme and Time-Stepping

Finite volume Euler methods of Jameson⁸ and Mavriplis and Jameson⁹, using fourth-order Runge-Kutta time-stepping, are adopted to interactively solve for the flows in different zones. These methods are briefly reviewed here to point out some modifications that are made in the present work.

The Euler equation for two-dimensional inviscid flow in integral form for a region Ω with a boundary $\partial\Omega$ is given as

$$\frac{\partial}{\partial t} \iint_{\Omega} w dx dy + \int_{\partial\Omega} (f dy - g dx) = 0 \quad (1)$$

where x and y are Cartesian coordinates and

$$w = \begin{vmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{vmatrix}; \quad f = \begin{vmatrix} \rho u \\ \rho v^2 + p \\ \rho uv \\ \rho uH \end{vmatrix}; \quad g = \begin{vmatrix} \rho v \\ \rho v^2 + p \\ \rho vH \end{vmatrix} \quad (2)$$

Here ρ , u , v , p , E and H are density, velocity components, pressure, total energy, and total enthalpy, respectively. For a perfect gas we have

$$E = \frac{p}{(\gamma-1)\rho} + \frac{u^2 + v^2}{2}, \quad H = E + \frac{p}{\rho}$$

Thus, we only need to solve for four variables: ρ , ρu , ρv and ρE . Equation (1) is discretized over individual control volumes (triangles) in a cell-centered approximation in which the flow variables are stored at the center of each cell. The above discretization procedure with the addition of artificial dissipation terms results in a set of ordinary differential equations.

$$S_1 \frac{dw_1}{dt} + [Q(w_1) - D(w_1)] = 0 \quad (3)$$

where S_1 is the area of the cell, Q is the spatial approximation of fluxes given by the second part of Eq. (1), and D is an appropriately constructed dissipation operator.

The fourth-order Runge-Kutta scheme is used to advance the solution in time from time step n to time step $n+1$. With the nonlinear operator P defined as

$$P(w) = \frac{1}{S} [Q(w) - D(w)] \quad (4)$$

we have

$$\begin{aligned} w(0) &= w^n \\ w(1) &= w(0) - \frac{\Delta t}{2} P[w(0)] \\ w(2) &= w(0) - \frac{\Delta t}{2} P[w(1)] \\ w(3) &= w(0) - \Delta t P[w(2)] \\ w(4) &= w(0) - \frac{\Delta t}{6} \{P[w(1)] + 2P[w(1)] \\ &\quad + 2P[w(2)] + P[w(3)]\} \\ w^{n+1} &= w(4) \end{aligned} \quad (5)$$

The artificial dissipation operator is calculated in the first and third step only to save computing costs.

In the region with structured grids implicit residual averaging is used to increase the base of the CFL number. It is shown¹⁰ that implicit smoothing increases the stability to Courant numbers much greater than the Courant number limit of the explicit scheme. The calculations presented in this work are obtained with a CFL number of 7. In addition, a variable time step, based on the maximum stability limit set by the local Courant number, and enthalpy damping are used to accelerate the convergence of the solution. Dissipation terms are formed as a blend of second- and fourth-order terms, coefficients of which are adapted to the flow. The resulting scheme is second-order accurate in the regions of smooth flow and first-order accurate near the shocks. To ensure that these dissipative terms are significant only in the vicinity of a shock, the second-order dissipation terms are scaled by the local Laplacian in the pressure.

Similar to the structured flow solver, in the unstructured flow solver fourth-order Runge-Kutta time stepping is also used to advance the solution in time. The support of the time-stepping scheme

here is, however, increased by an explicitly residual averaging scheme. If the residuals at cell 1 are

$$R_1(w) = \frac{1}{S_1} [Q(w_1) - D(w_1)]$$

they are replaced by

$$\overline{R}_1 = \epsilon R_1 + \frac{1-\epsilon}{3} \sum_{k=1}^3 R_{1k}$$

where R_{1k} are residuals at three forming points of the triangular cell. The residuals at nodal points are obtained as the average of residuals at all cells having that point in common. ϵ is a constant which is chosen as 0.6. With this smoothing scheme, the CFL number for the unstructured flow solver could also be increased to about 7.

It is found that the CFL number in the two zones must be similar to facilitate the convergence of the solution in the entire domain.

4.0 Dissipative Terms

The structured flow solver uses a blend of second- and fourth-order dissipation terms to prevent odd and even decoupling of the solution. The artificial dissipation for the unstructured flow solver is similarly constructed as a blend of undivided Laplacian and biharmonic operators. Generally, in the absence of shocks in subsonic flows, only biharmonic dissipation is required. However, it is found that in regions of large pressure gradients such as at the leading edge of the flap, second-order dissipation terms are still needed even in the subsonic flow regime.

To obtain the fourth-order dissipation term in the triangular mesh zone, an undivided four-point Laplacian operator is first defined as

$$\nabla^2 w_1 = \sum_{k=1}^3 w_k - 3w_1$$

where w represents the flow variables ρ , ρu , ρv , and ρh . The dissipation flux across a cell face, $1k$, delimiting cells, 1 , and its neighbors, k , are then calculated as

$$d_{1k} = \epsilon_{1k}^{(4)} A_k (\nabla^2 w_1 - \nabla^2 w_k) \quad (6)$$

where

$$A_k = \left| u_k \Delta y_k - v_k \Delta x_k \right| + c_k \sqrt{\Delta x_k^2 + \Delta y_k^2}$$

Here Δx_k and Δy_k are coordinate increments of the edge and u_k , v_k , c_k are velocity components and the speed of sound along the edge, respectively, and are taken as the average of the values at cells 1 and k . The A_k term, which is proportional to the size of the cell face, k , and represents the maximum eigenvalue of the Euler equation in the direction normal to the face, scales appropriately with the time derivative in Eq. (3). In order to more accurately scale the cells which have higher aspect ratios, A_k is not integrated around the boundary of the control volume, as is commonly done. The $\epsilon_{1k}^{(4)}$ is a constant defined later.

The second-order dissipation term is similarly constructed by replacing $\nabla^2 w_1$ and $\nabla^2 w_k$ in Eq. (6) with w_1 and w_k , respectively, so that we have

$$d_{1k} = \epsilon_{1k}^{(2)} A_k (w_1 - w_k)$$

where $\epsilon_{1k}^{(2)}$ should be of order one near a shock and of order $(\Delta x)^2$ in regions of smooth flow to preserve the second-order accuracy of the scheme. To ensure this, $\epsilon_{1k}^{(2)}$ is then scaled proportional to an undivided Laplacian in the pressure

$$\epsilon_{1k}^{(2)} = \epsilon_2 \frac{\left| \sum_{k=1}^3 p_k - p_1 \right|}{\sum_{k=1}^3 p_k - p_1}$$

It is found, however, that a better scaling factor for the edges that are on the solid boundary is the pressure gradient in the surface-wise direction. Thus, the pressure gradient term for cells on the solid boundary is defined here as

$$\nu_1 = \left(\frac{p_{i+1} - 2p_1 + p_{i-1}}{p_{i+1} + 2p_1 + p_{i-1}} \right)$$

where i , $i-1$, and $i+1$ are the adjacent cells to cell 1 , which are on the surface boundary. For these cells, $\epsilon_{1k}^{(2)}$ is then taken as

$$\epsilon_{1k}^{(2)} = \epsilon_2 \max(\nu_{i-1}, \nu_1, \nu_{i+1})$$

It is known that the fourth-order dissipation term may produce overshoots in the vicinity of a shock; therefore, they are turned off by defining

$$\epsilon_{1k}^{(4)} = \max[0, \epsilon_{1k}^{(4)} - \epsilon_{1k}^{(2)}]$$

Here ϵ_2 and ϵ_4 are empirically determined constants.

The final form of the dissipative flux is then

$$d_{1k} = \epsilon_{1k}^{(2)} A_k (w_1 - w_k) + \epsilon_{1k}^{(4)} A_k (\nabla^2 w_1 - \nabla^2 w_k)$$

This dissipation flux, d_{1k} , is added or subtracted from its adjacent cells consistent with the direction of the normal to each edge. This ensures that dissipation is conserved throughout the region.

The treatment of the boundary conditions is known to affect the convergence and accuracy of the solution. The wall pressure can be extrapolated from the pressure at the center of the boundary cell using the method given in Ref. 8. However, in the present calculations it is found, in both the structured and unstructured solvers, that the normal pressure gradient, $\partial p / \partial n$, is negligible, and the pressure of the boundary could be assumed equal to the pressure at the center of the boundary cell.

It is important to note that when the solid boundary condition is imposed in the unstructured flow solver, however, is that setting the normal components of the fluxes to zero and only accounting for the pressure terms in the momentum equation does not necessarily satisfy the flow tangency condition on the boundary. A stronger

formulation is needed to ensure this requirement. In the present cell-centered scheme, a stronger form of the boundary condition is imposed in order to compute the artificial dissipation terms more exactly. Flow variables ρ , ρu , ρv , H in imaginary cells inside the solid surface are extrapolated based on the assumptions of no normal flux and equal tangential fluxes between cells outside and inside the boundary. These values are then used in the calculation of dissipation terms associated with the edges that are on the boundary. This can be shown to be equivalent to explicitly setting velocities to be tangent to the wall, as suggested in Ref. 11.

5.0 Zonal Interference Scheme

Interaction between neighboring zones could be greatly simplified by ensuring that neighboring structured and triangular grids have complete, common, edges. This could easily be done along the zonal boundaries by choosing the structured grid points as forming points of the triangular grids. Mass, momentum and energy fluxes through the zonal boundaries are conserved by using the values from neighboring zones when calculating fluxes. This requires maintaining information on the grid interaction, including indexes of structured grids next to the boundary triangular meshes, and vice versa.

Careful attention must be given to calculation of dissipation terms for the boundary cells in order to ensure conservation of dissipation terms throughout the flow. Dissipation terms for the triangular, boundary cells are calculated by extracting information from both the structured and unstructured zones and vice versa. Inappropriate treatment of dissipation terms can result in solution inaccuracy along the zonal boundaries or can produce considerable decoupling of the solution in boundary regions of both zones and contaminate the solution in the entire region. It is also found that comparable CFL numbers and degrees of smoothing in different zones, are essential to improve the overall convergence of the solution in the entire region.

6.0 Results

The multielement airfoil of Fig. 1 with overhang of 5%, gap-to-chord ratio of 10%, and flap angle of rotation of 15° is considered as the first computational example. The isobar solution at a Mach number of 0.2 is shown in Fig. 2, which indicates a smooth solution across the zonal boundaries. This solution is obtained by using 256×320 -meshes, part of which are replaced by 976 triangular meshes. A CFL of 7, along with a variable time step based on a maximum limit set by local Courant number, is used in both zones.

Figure 3 shows the details of the solution in the unstructured zone and indicates that the strong form of the boundary condition discussed earlier in Section 4.0 adequately satisfies the flow tangency requirements on the solid boundary. The solution also accurately predicts the location of the stagnation point on the leading edge of the flap. Furthermore, even though the equations are inviscid, the addition of dissipation terms introduces viscous flow like vorticity in the flap well.

Calculated surface pressures for both the main airfoil and the flap are compared with experimental data of Ref. 12 in Fig. 4(a,b). The experimental results are obtained at a $M = 0.195$ and Reynolds number of 500,000 based on unextended chord length. The comparison is obviously very poor in the flap well region due to the viscous effects which are dominant in this region. Our inviscid model also is unable to predict the separation that occurs on the upper surface of the flap as indicated by the experimental data.

Our second computational example deals with iced airfoils. The analysis of the aerodynamic performance of iced airfoils has been of great interest to aircraft designers. In order to find ways to prevent ice formation on wings, one needs to accurately predict the flowfield about iced airfoils with various forms of ice shapes.¹³ Ice accretion on an airfoil produces very irregular and rough surfaces on the leading edge region. These shapes normally have concavities and convexities which cannot be modeled using a single zone structured grid.

An iced NACA 0012 airfoil with an eight-minute ice surface computed using the fortified Lewice program of Ref. 14 is shown in Fig. 5. A composite structured and unstructured grid is generated consisting of 198×320 -meshes, part of which are replaced by 932 triangular meshes. The unstructured mesh region is extended far enough to cover the irregular ice shape on the leading edge. A converged solution obtained using this mesh is presented in subsequent figures.

Figure 6 shows the isobar solution obtained at $M = 0.2$ and $\alpha = 4^\circ$. The compressibility effects at this Mach number are less than 2%, according to the Prandtl Glauert rule. The contour lines vary smoothly across the zonal boundary indicating that the conservation of fluxes is well satisfied.

Figure 7 shows the velocity vectors in the leading-edge region. This figure clearly indicates that there are multiple stagnation regions where streamlines approach the surface from the farfield and are divided in two opposite directions on the surface. There is also a small region of reverse flow on the upper surface. Similar to the first example, although the equations are inviscid, the addition of dissipation terms produces viscous flow like vorticity behind the ice shape on the upper surface.

The computed pressure distribution on the surface is compared with the surface panel method results in Fig. 8; good agreement is indicated.

7.0 Conclusion

The objective of the present work is to develop an efficient and reliable two-dimensional zonal approach capable of coupling structured and unstructured grid zones for complex airfoil shapes. This approach requires considerably less memory compared to using an entirely unstructured mesh capable of handling complex geometries. It also takes advantage of the different desirable features of structured and unstructured meshes. Computational examples are presented that demonstrate the applicability of this method to the analysis of the flowfields such as those around

iced airfoils, multielement airfoils, and blunt and divergent-trailing-edge airfoils. The method could also be coupled with a boundary-layer method to include the viscous effects.

8.0 References

1. Batina, J.T., "Unsteady Euler Algorithm with Unstructured Dynamic Mesh for Complex-Aircraft Aerodynamic Analysis," AIAA Journal, Vol. 29, 1991, p. 37.
2. Roe, P.R., "Error Estimates for Cell-Vertex Solution for the Compressible Euler Equations," ICASE Report No. 87-6, NASA CR-178235, 1987.
3. Mavriplis, D.J., "Accurate Multigrid Solution of the Euler Equations on Unstructured and Adaptive Meshes," AIAA Journal, Vol. 28, 1990, p. 213.
4. Venkatakrishnan, V., Mavriplis, D.J., "Implicit Solvers for Unstructured Meshes," AIAA Paper 91-1537-CP, Honolulu, Hawaii, 1991.
5. Venkatakrishnan, V. and Batina, J.T., "Application of Direct Solvers to Unstructured Meshes for the Euler and Navier-Stokes Equations Using Upwind Schemes," AIAA Paper 89-0364, 1989.
6. Chin, V.D., "2-D Unstructured Grid Generation," M.S. Thesis, California State University, Long Beach, 1991.
7. Bowyer, A., "Computing Dirichlet Tessellation," The Computer Journal, Vol. 24, No. 2, 1981.
8. Jameson, A., Schmidt, W. and Turkel, E., "Numerical Solution of the Euler Equation by Finite-Volume Method Using a Runge-Kutta Time-Stepping Scheme," AIAA Paper 81-1259, 1981.
9. Mavriplis, D.J. and Jameson, A., "Multigrid Solution of the Two-Dimensional Euler Equation on Unstructured Triangular Mesh," AIAA Paper 87-0353, 1987.
10. Jameson, A. and Baker, J.J., "Solution of the Euler Equation for Complex Configurations," AIAA Paper 83-1929, 1983.
11. Mavriplis, D.J., "Accurate Multigrid Solutions of the Euler Equation on Unstructured and Adaptive Meshes," AIAA Journal, Vol. 28, 1990.
12. Chen, H.H., Private communication.
13. Shaw, R.J., Potapczuk, M.G. and Bidwell, C.S., "Prediction of Airfoil Aerodynamic Performance Degradation Due to Icing," Numerical and Physical Aspects of Aerodynamic Flows, IV (ed. T. Cebeci), Springer-Verlag, 1990.
14. Cebeci, T., Chen, H. and Alemdaroglu, N., "Fortified Lewice with Viscous Effects," Journal of Aircraft, Vol. 28, No. 9, pp. 564-571, Sept. 1991.

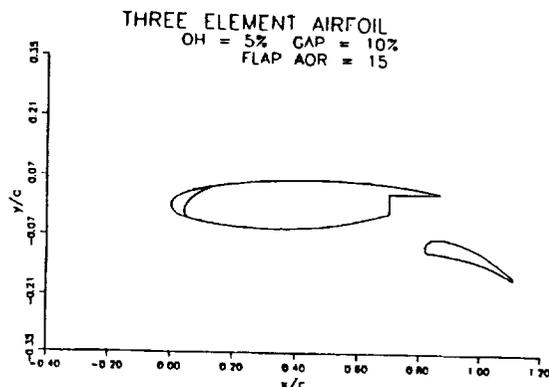


Fig. 1. Multielement airfoil with large flap deflection.

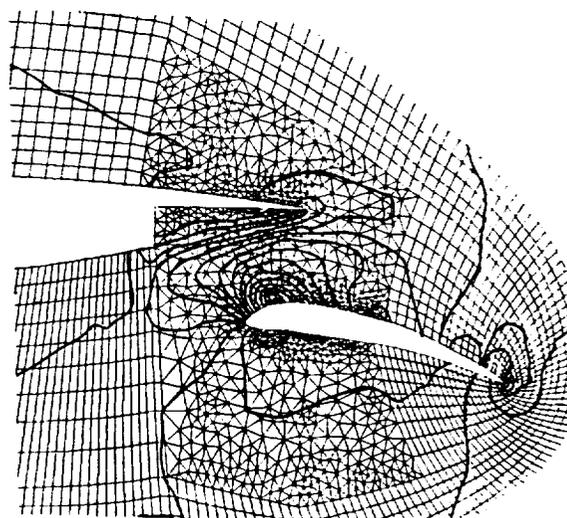


Fig. 2. Isobar solution for multielement airfoil of Fig. 1 using composite grid. $M = 0.125$, $\alpha = 0$.

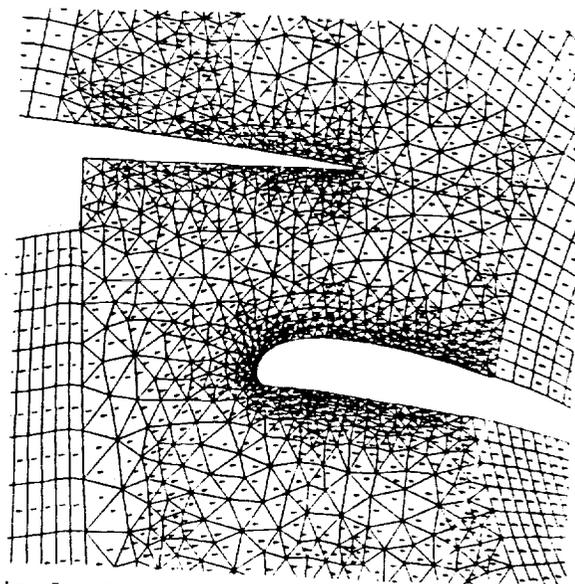


Fig. 3. Total velocity direction for multielement airfoil of Fig. 1 using composite grid $M = 0.125$, $\alpha = 0$.

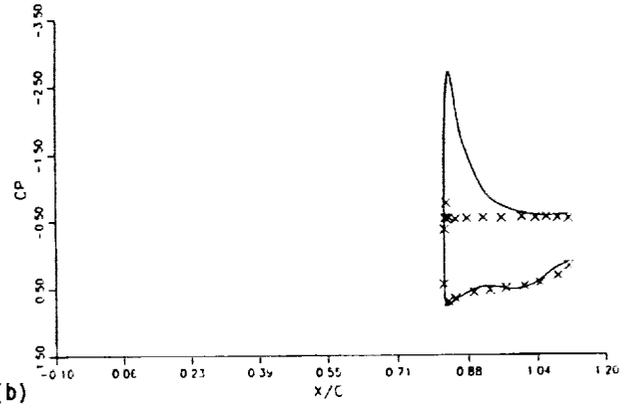
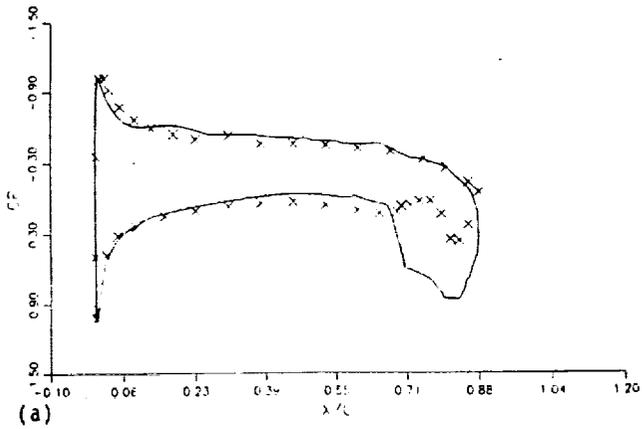


Fig. 4. Comparison of calculated C_p with experimental results; (a) main airfoil, (b) flap.

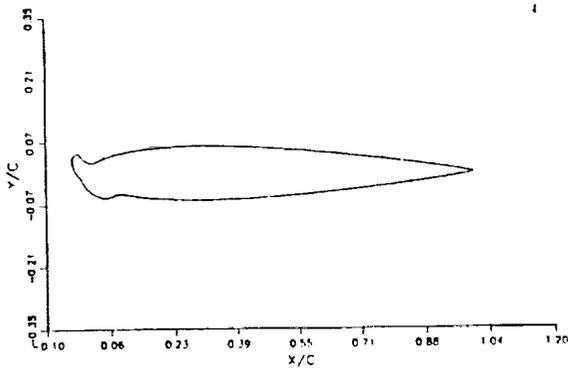


Fig. 5. Eight minute ice shape on a NACA 0012 airfoil at $\alpha = 4^\circ$.

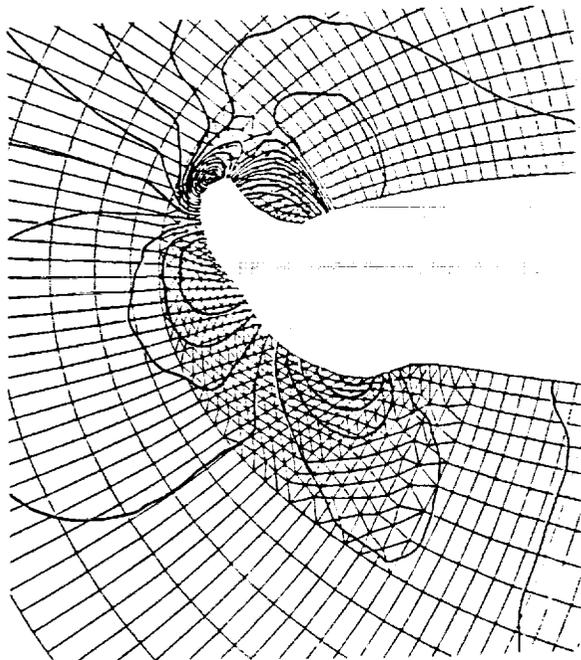


Fig. 6. Isobar solution obtained using composite grid, $M = 0.2$, $\alpha = 4^\circ$.

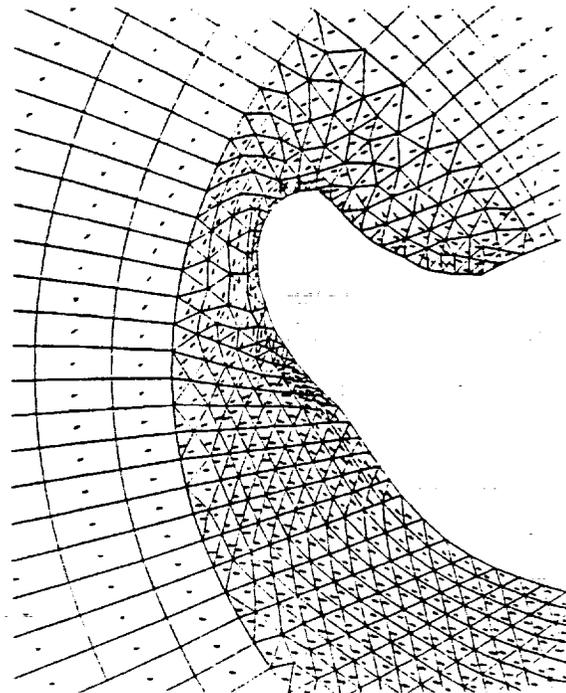


Fig. 7. Total velocity vectors for iced airfoil of Fig. 5 using composite grid.

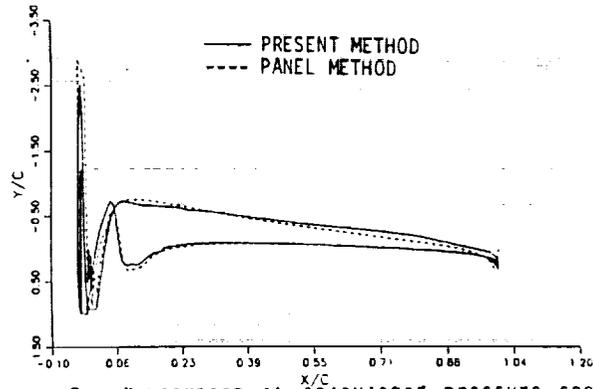


Fig. 8. Comparison of calculated pressure coefficient with panel method calculation.